Neural Computing and Applications (2023) 35:14577–14589
https://doi.org/10.1007/s00521-021-06118-z

S.I: ML4BD_SHS

An improved deep learning mechanism for EEG recognition in sports health informatics

Tong Zhao[1] · Jingyi Zhang[1] · Zuocan Wang[2] · Ryan Alturki[3]

Received: 29 January 2021 / Accepted: 11 May 2021 / Published online: 25 September 2021

Abstract

A growing number of studies indicate that concussed athletes may have long-term residual electroencephalography (EEG) defects that can last up to ten years after the injury. With the use of conventional concussion screening techniques, these abnormalities are often ignored. As a result, returning to sports earlier can result in recurrent concussions, raising the risk of recurrent concussions with more severe consequences. This study uses deep learning methods to analyze the EEG signals of athletes. It then proposes and designs a channel attention module connected to the input layer of the convolutional neural network (CNN). The proposed approach automatically learns the EEG signals of different channels for recognizing the contribution of the task. The CNN is then connected to the recurrent neural network (RNN) for further processing. Based on this approach, this study combines the residual unit and the channel attention model to propose a convolutional recurrent neural network (CRNN) structure that is highly effective for EEG signal recognition. In this study, the EEG dataset of the Stanford research project has been used for experimental analysis. The performance of the proposed scheme is evaluated with the help of various performance measures. The experimental result shows that the proposed model improves the recognition accuracy from 82.58% of ResNet13 to 85.68% and attained excellent recognition accuracy of 91.05% by using CAMResNet13 + CRNN architecture.

Keywords Deep learning · EEG signal · Convolutional neural network (CNN) · Recurrent neural network (RNN) · Health informatics

✉ Zuocan Wang
wangzuocan@lyu.edu.cn

Tong Zhao
2111516032@e.gzhu.edu.cn

Jingyi Zhang
zpsheng163@163.com

Ryan Alturki
rmturki@uqu.edu.sa

1 School of Physical Education, Henan University of Chinese Medicine, Zhengzhou 450046, Henan, China

2 School of Sports and Health, Linyi University, Linyi 276000, Shandong, China

3 Department of Information Science, College of Computer and Information Systems, Umm Al Qura University, Makkah, Saudi Arabia

1 Introduction

In recent decades, EEG signal analysis [1, 2] and pattern recognition have been a hot issue in biomedical engineering field. EEG signals and their derived biological event-related potentials are widely used in neuroscience, cognitive science, cognitive psychology, and medicine. More and more studies have shown a correlation between EEG signals and behavior recognition tasks; for example, people's intentions can be recognized through EEG signals. Therefore, EEG signal analysis and pattern recognition research technology are also promising technology. Simultaneously, with the progress of science and technology in recent years, deep learning technology has been rapidly developed and widely used. Given the powerful feature learning and nonlinear fitting capabilities of deep learning, it has been applied to EEG signal analysis and pattern recognition research and has achieved outstanding results [3–5].

In neuroscience, EEG signals can be used in clinical neurology to study the brain and diagnose many neurological diseases, such as the use of EEG signals to detect epilepsy diseases automatically. As a common chronic brain disease, epilepsy causes great harm to the patient's body and mind. The brain is abnormally excited during an epileptic attack, usually manifested as behavioral changes, such as loss of consciousness, erratic movements, temporary loss of breathing and memory, among other symptoms. At present, the primary method of diagnosing epilepsy is that doctors and experts diagnose whether there is an epileptic seizure by observing the recorded EEG signal contains epileptic carbuncle waves different from normal signals [6–10]. Due to the random nature of epilepsy, long-term real-time monitoring of the patient's EEG signals is often required, which leads to lengthy detection time and low detection efficiency. Therefore, the automatic detection of epilepsy diseases based on EEG signals is of great help in reducing the workload of medical workers. Besides, EEG signals are also used to diagnose other neurological diseases, such as Alzheimer's disease, brain tumors, stroke, and Parkinson's disease. In psychology, EEG signals are also beneficial. For example, using EEG signals to study consumers' psychology and customizing prices according to changes in EEG signals when consumers see different prices to be more in line with their psychology. Another example is the use of EEG signals to analyze the mental activities of criminals, so that when the criminals are interrogated, the mental activities of the criminals can be inferred from the changes in the criminals' EEG signals to formulate more precise interrogation plans.

In terms of cognitive science, we can use EEG signals to detect the driver's driving state [11–15], such as whether the driver is tired, whether the attention is concentrated, etc., and then prompt the driver according to the driver's state. We can also use EEG signals to identify the patient's emotional state to treat him/her better and improve the treatment effect. Especially for those with autism and depression who are unable or unwilling to express their emotions, it is essential to obtain the emotional state of these patients and choose more scientific treatment methods. Besides, students can obtain their learning status in real-time through EEG signals. For example, we can determine if their attention is focused, whether their mood is good, etc., and then adjust learning strategies in time according to these learning statuses, thereby improving their learning efficiency.

In medicine, brain-computer interface technology can be used to allow those patients who have lost part of their motor abilities to control wheelchair walking, make calls, and control prosthetics and spelling through EEG signals by improving their quality of life. The application of the aforementioned EEG signals in various aspects involves the critical issue of EEG signal recognition. Hence, EEG signal analysis and pattern recognition research will be the key to promoting EEG signal applications. In recent decades, with the continuous efforts of researchers, EEG signal analysis and pattern recognition technology have been rapidly developed [16]. However, in recent years, with the deepening of research, researchers have found significant bottlenecks in EEG signal analysis and pattern recognition based on feature engineering and traditional machine learning methods, limiting the broad application of EEG signals in various fields. At the same time, deep learning technology has developed rapidly and has achieved remarkable results. Given the powerful automatic feature extraction capabilities and nonlinear fitting capabilities of deep learning methods, they are expected to break through traditional machine learning's bottlenecks. It also shows that EEG signal analysis and pattern recognition technology research, based on deep learning methods, are of great significance to applying EEG signals in various aspects.

At present, the recognition of EEG signals mainly relies on traditional machine learning methods. However, the performance of these methods primarily depends on feature design and different recognition tasks that require various features to be designed. Therefore, it is difficult for traditional machine learning methods to be universal in EEG signal recognition. Fortunately, in recent years, deep learning methods as representation learning have developed rapidly and have achieved good results in many fields. Deep learning can automatically learn useful EEG signals from raw data to perform the pattern recognition process, which is especially suitable for EEG signal recognition tasks. Hence, numerous EEG signal recognition methods based on deep learning have appeared in recent years. However, these methods have some shortcomings. For example, the contribution of different channel information of multi-channel EEG signals to the recognition task is not considered in the application process; for example, the time-series information of brain is not used well in the application process. This paper has conducted research to solve such issues and put forward some methods to solve the above problems.

The rest of this paper is organized as follow. In Sect. 2, related work is presented. In Sect. 3, the proposed methodology and the datasets are discussed by proposing an improved Deep Learning mechanism for EEG Recognition. The experimental results and analysis are comprehensively discussed in Sect. 4. Finally, the paper is concluded and future research directions are provided in Sect. 5.

2 Related work

Earlier, the BCI system was mainly composed of five stages which include: (1) data acquisition in the form of EEG signal, (2) pre-processing of data, (3) extraction of useful features from the data, (4) classification based on the provided features, (5) interface for controlling the device [17]. The first stage is used to collect EEG signals, convert these signals to digital format and stored it in the system. The second stage is used to remove the noise from data, transformation of data, and clean the raw data. The third stage is used to extract useful features from the EEG signals that contain valuable information and are used for further processing. The fourth stage describes the classification of the extracted features via various ML classification algorithms. The fifth and final phase, i.e., device control is used to control the device based on the categorized signals, and these signals are then converted to different commands which then help in the controlling of various devices [18–23].

In the last couple of years, numerous research studies have been conducted that mainly focuses on the extraction of features and classification and lifted a valuable impact on the performance of the BCI system [24–26]. Various feature extraction algorithms are used over the years for extracting promising features from the raw EEG signals; CSP is one of them and is considered to be a powerful and classical algorithm. Bharathan et al. [27] have proposed a method based on two CSP approach for EEG classification and attained promising results. Chin et al. [28] proposed a filter-bank CSP technique and achieved significant classification performance. The other important feature extraction techniques used widely are, CWT, STFT, and EMD [29, 30]. Tabar et al. [29] proposed a system based on STFT feature extractor that extracts features from raw EEG signals and finds the location, time, and frequency information which is then converted to the images using image segmentation. Lee et al. [30] extract time frequency spectrum with the help of CWT. For increasing the computation performance, they used feature selection techniques in order to avoid the duplicate information. Various machine leaning techniques like LDA, PCA, SVM, and ANN, etc., are used frequently for the features classification [31, 32]. Naseer et al. [31] have used LDA for the classification of MI problem that was based on two features. Siuly et al. [32] proposed a system for the classification of binary class MI signals, using the SVM classification algorithm. The conventional approaches used for the classification of EEG signals depend on the expertise and previous knowledge of EEG signal processing.

With the passage of time, deep learning techniques starts showing its ability of automatic extraction of features from the data and replaced the traditional approaches. Various deep learning techniques have been used for the classification of EEG signals, and it showed good performance as compared to the previous approaches. Among the deep learning models, CNN is the most widely used model for the EEG signal classification. Yang et al. [33] proposed a scheme for the classification of multiclass MI EEG signals based on CSP feature extractor and the CNN model and attained the classification accuracy of 69.72%. Though, CNN has showed significant contribution in the classification of EEG signals, but still it lacks some important considerations. For example, CNN is the go to and the most precise model for the natural language processing (NLP) problems, image classification, and biological data, etc., but a few researchers have used it for solving the problem of EEG signal classification because training a new CNN model from the scratch requires a large amount of EEG labeled data. In addition, in practical applications the EEG data are mostly unlabeled, and to label these signals data manually is a big headache which is almost impossible. Another reason is that, having a small amount of data for training leads to overfitting problem and training a new deep CNN model from the scratch is very time consuming and computational intensive problem.

In order to solve the problems in the previous approaches as mentioned above and to improve the classification results, this study uses deep learning classification algorithms to classify the EEG signals in an efficient manner.

3 Material and method

The material used and the methods followed in carrying out this research study are represented in this subsection.

3.1 Dataset

The development of an automated and intelligent system extensively depends on the problem related dataset. It means that problem specific dataset has a very high influence on the efficiency of an intelligent model. Considering the significance of the dataset, this study utilized the EEG dataset of Stanford research project for all the experimental analysis. Complete information and description of the utilized dataset is available in [34].

3.2 Proposed methodology

The main goal of the proposed system is to identify the health conditions of an athlete through his/her EEG signals. In this study, two deep learning algorithms are investigated to classify the EEG signals. Various preprocessing techniques are used to provide the data in a normalized form to the classification models. To measure the performance of each classification model, various performance measures are investigated. Figure 1 shows the block diagram of the proposed system.

3.3 Data preprocessing

Data preprocessing is an important technique used for representing the data in an organized manner to the classification models. In this study, numerous preprocessing techniques like, MinMax scaler, standard scaler, and noise removal are used to make the dataset more efficient for the classification.

3.4 Utilized Deep Learning algorithms

In this study, two deep learning algorithms were utilized for the classification of EEG signals in order to measure the health conditions of an athlete. The first algorithm utilized was LSTM, while the other one was a new proposed algorithm known as CRNN. The reason behind using more algorithms is that the importance of a model varies with the nature of the problem because in advance nobody knows that which algorithm will best suit on my problem. Using several models mean to select the most generalize and reliable model for EEG signal classification. A brief introduction to the utilized DL algorithms is given below:

3.4.1 Long Short-Term Memory (LSTM)

The basic recurrent neural network (RNN) has the problem of gradient explosion and disappearance during the back propagation process, which will cause the current time information to be unable to be transmitted over long distances. Therefore, the basic recurrent neural network is difficult to deal with long-distance dependencies. But fortunately, the LSTM solves the long-distance dependence problem through some special internal structures. It has been widely used in speech recognition, image description, natural language processing, and other fields and has achieved good results. All recurrent neural network structures can be seen as stacks of identical structural units (neural networks) in the horizontal direction. In a basic RNN, this structural unit is very simple, such as only a tank layer, as shown in Fig. 2. The LSTM network also has a similar structure, but the repeating module has a different structure. As shown in the middle part of Fig. 3, it is no longer a simple tank layer, but is composed of four parts interacting. In this Figure, the arrowed line represents the flow direction of the vector, the circle represents the arithmetic operation between vectors, such as the element-wise addition operation between vectors, etc., the box represents an activation function, and the merging of lines. The represented vectors are merged, and the lines are separated to show that the vectors are copied to the other places.

The key to the long-term dependence of the LSTM network is the state of the unit and the horizontal line on the unit that spans the entire structure. The horizontal line

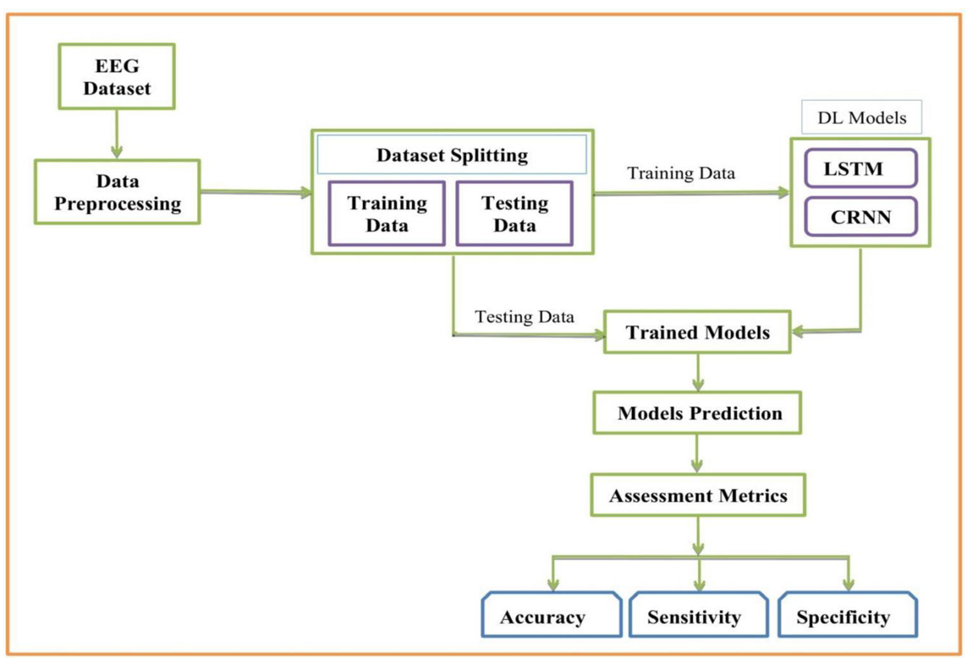

Fig. 1 Block diagram of the proposed system methodology

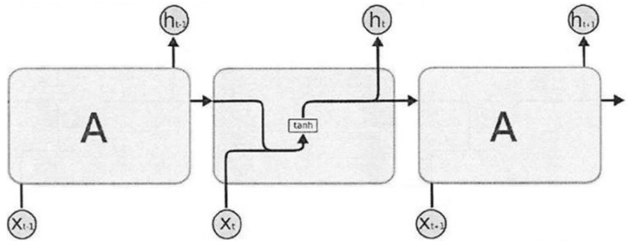

Fig. 2 Structural unit diagram of cyclic neural network

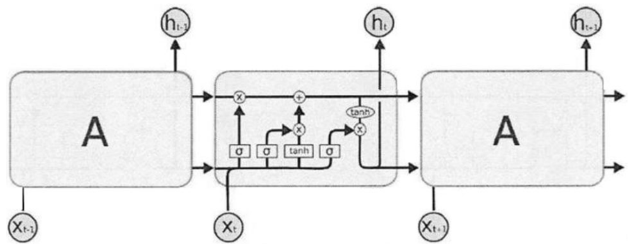

Fig. 3 LSTM structural unit diagram

on the unit is like a conveyor belt through which the vector traverses the entire unit with only a few operations. Therefore, this structure can easily realize that information is transmitted along it without changing. At the same time, the unit structure can achieve selective information passing through a structure called a gate. LSTM has three such gate structures, namely forget gate layer, input gate layer, and output gate layer. The gate structure is a way to choose to pass information. They are composed of S-shaped neural network layers and point-by-point multiplication. Sigmoid layer is used at the last layer which output a value between 0 and 1. A value of zero means "let nothing pass," and a value of 1 means "let everything pass".

Long short-term memory network (LSTM) not only has the ability to characterize the context of time series, but also solves the problem of long-distance dependence of time series. Therefore, long short-term memory network is often used in speech recognition, video classification, natural language processing, and other fields. And a certain breakthrough has been made. As a typical time series signal, EEG signals are considered to be a very reasonable idea for applying long and short-term memory networks to EEG signal recognition. However, previous experiments have shown that EEG signals are directly sent to long-term short-term memory network has not achieved good results. The reason for the analysis may be that the EEG signal is weak and easy to be disturbed by noise during the acquisition process, resulting in a low signal-to-noise ratio of the EEG signal. The long-short-term memory network is mainly used to learn before and after the sequence associated information, and it is difficult to extract effective features, which leads to poor recognition. In order to solve this problem, we connect the CAMResNetl3 network designed in Chapter 4 to the front layer of the long- and short-term memory network. This has the following advantages. For the memory network, the CAResNet13 network can be used to extract features with more characterization capabilities to establish the contextual information of the EEG signal. For CAResNet3, it solves the ability of CAResNet13 to learn the time series information of the EEG signal. The experimental result also shows that the result of the combination of the two is better than the result of only one of the methods.

3.4.2 Convolutional Recurrent Neural Network (CRNN) Architecture

CRNN is a combination of both the CNN and RNN and is being used for the classification of EEG signals here in this study. The network structure of the RCNN is shown in Fig. 4. It is composed of four parts from bottom to top, namely the convolution layer, the circulation layer, the attention layer, and the classification layer. The convolutional layer is also called the feature extraction layer. It mainly uses the powerful feature extraction capabilities of the convolutional neural network to automatically extract features from the input original EEG signal and then send the extracted feature sequence to the RNN layer. The main function of the RNN layer is to learn the before and after associated information of the EEG signal through the feature sequence and then send the feature vector output by the RNN layer at each moment into the Attention layer. The main function of the Attention layer is to automatically learn the weighted feature vectors from the RNN layer, should give at each moment, and then sum these feature vectors to obtain the weighted and summed feature vectors and send them to the classification layer for classification. The network has the following characteristics: (1) there is no need to manually design features, and useful information can be learned directly from the original EEG signals, (2) It can process variable-length EEG signals and make full use of the related information before and after EEG signals, (3) end-to-end learning is possible.

3.4.2.1 Feature extraction layer (CNN layer) structure Considering that the EEG signal is very weak, it is very susceptible to the influence of external noise during the collection process, resulting in a low signal-to-noise ratio of the collected EEG signal. This is also the reason why the direct use of RNN to classify EEG signals is not effective. The natural idea is to extract useful feature sequences from EEG signals before applying RNN, and then send these feature sequences to RNN, and let RNN learn the correlation information before and after EEG signals from these feature sequences. Different from

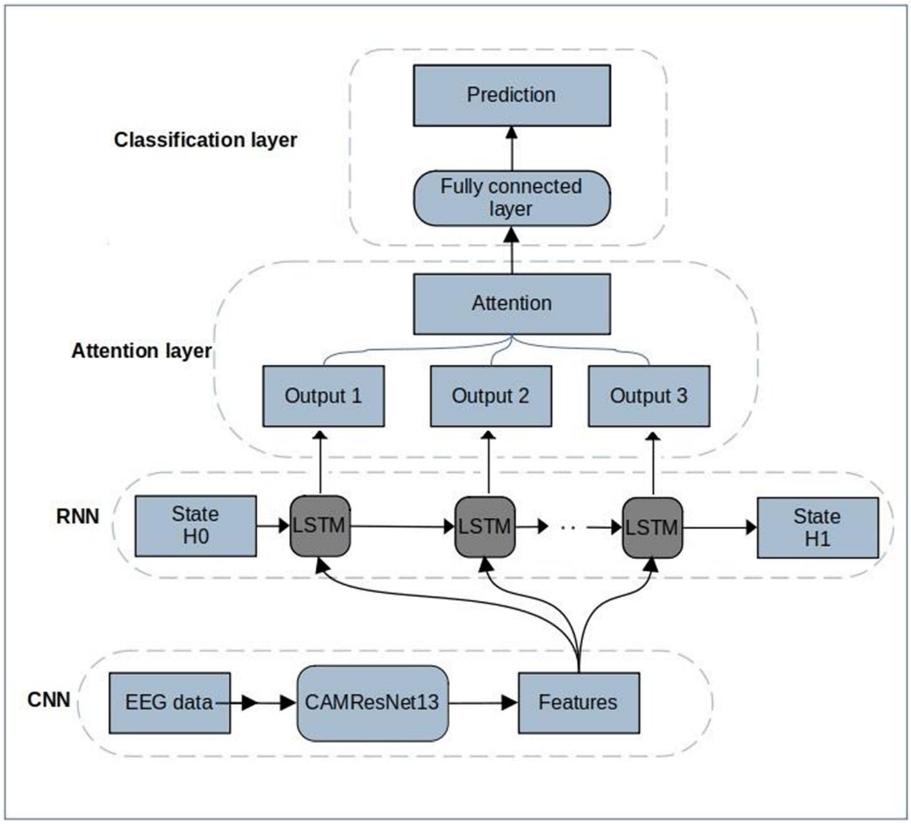

Fig. 4 Proposed Convolutional Recurrent Neural Network (CRNN) diagram

artificial design features, this paper uses the convolutional neural network CAResNet 13 designed to automatically extract useful EEG signal features. The detailed process is

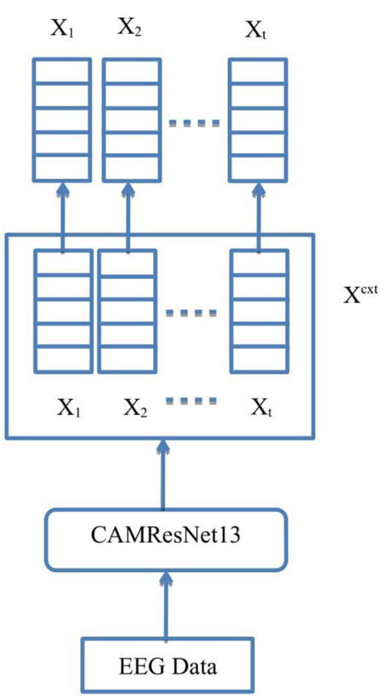

Fig. 5 Structure diagram of feature extraction layer

shown in Fig. 5. First, we apply CAResNet 13 (with the following AvgPool, FC and Softmax layers removed) to extract the features of the original input EEG signal and obtain the output feature vector x, whose time dimension is t, and the channel size of the dimension is c. After that, we slice the output feature vector along the time dimension to get the feature vector sequence $x = [x_1, x_2, \ldots, x_t]_{c\times t}$, that is, the feature vector sequence is composed of t vectors of dimension c. Each feature vector is associated with a receptive field. It can also be considered that each feature vector is a descriptor of the corresponding area of the original EEG signal.

Considering that the time dimension of the EEG data set used in this article is 32, if you follow the setting of CAMResNet13, every two residual channel attention units will sample the input by 2 times, and the input needs to be the data that are sampled 8 times. In this way, the time dimension of the output feature vector of CAMResNetl3 becomes 4, that is, the length of the obtained feature vector sequence is 4, which leads to the too short duration of the feature vector sequence sent to the RNN, which will inevitably affect the recognition accuracy of the network. Therefore, we change to only sample the input by 2 times in the fourth residual channel attention module, and the total sampling multiple of the input data for the entire network is 2. In this way, the time dimension of the output feature vector of CAResNet13 is 16, and the length of the obtained feature vector sequence is also 16. The following experimental results also show that a total of 2 times sampling the input network has the best recognition effect on EEG signals. Other parameters of CAMResNet13, such as convolution kernel size k, compression rate r, etc., are all configured using the best experimental results.

3.4.2.2 RNN layer structure The function of the RNN layer is to use the feature vector sequence extracted by the feature layer to learn the correlation information before and after the EEG signal, and then output a feature vector $x = [x_1, x_{2,}..., x_t]_{c\times t}$ at each moment. The detailed process is shown in Fig. 6. First, the feature vector sequence from the feature extraction layer is sequentially sent to the input of LSTM, and then a vector x_t is input for each time of LSTM, and a feature vector h_t is also output, and then output at that time. The feature vector h_t and the input vector x_t are summed element by element to obtain the final output vector y_t of the RNN layer at that moment. The process formula is expressed as follows:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{1}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{3}$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{4}$$

$$o_t = \sigma(W[h_{t-1}, x_t] + b_0) \tag{5}$$

$$h_t = o_t * \tanh(C_t) \tag{6}$$

$$y_t = h_t + x_t \tag{7}$$

where x_t is the input vector of the RNN layer at time t, y_t is the output vector of the RNN layer at time t, W_f, W_i, W_o are the weights of the LSTM forget gate, input gate, and output gate, respectively, and h_{t-1} and C_{t-1} are LSTM t -1 output and unit state.

3.4.2.3 Attention layer structure For the feature vector $y = [y_1, y_{2,}..., y_t]_{c\times t}$ output at each moment of the RNN layer, one approach is to directly average these feature vectors to obtain the summed feature vector and use it as the input of the classifier. The prerequisite for this is that we believe that the feature vector output by the RNN layer at each moment is equally important for the classification of EEG signals. Obviously, this assumption is difficult to hold. For example, compared with the feature vector y_t output at the last moment of the RNN layer and the feature vector y_1 output at the first moment of the RNN layer, y_t encodes the information of the input feature vector at the previous moment, so in the EEG the role played in the signal recognition task will be relatively important. Therefore, how to assign different weights to the output vector of the RNN layer at each moment is particularly important for improving the classification effect of EEG signals. In order to allow the network to adaptively assign different weights to the output vector at each moment of the RNN layer, we design an attention model (attention) to solve this problem. The detailed structure is shown in Fig. 7. First, the learnable variable $\lambda = [\lambda_1, \lambda_{2,}..., \lambda_t]$ is obtained through the attention weights module, and then we normalize the variable λ through the softmax function to obtain the weight coefficient vector $w = [w_1, w_{2,}..., w_t]$ of each input vector and finally use these weighting coefficients to the input vector Perform weighted summation to obtain the final output vector, which will then be sent to the classifier for classification. In this process, the variable parameter setting is obtained by optimizing learning during back propagation. The formula of the process is as follows:

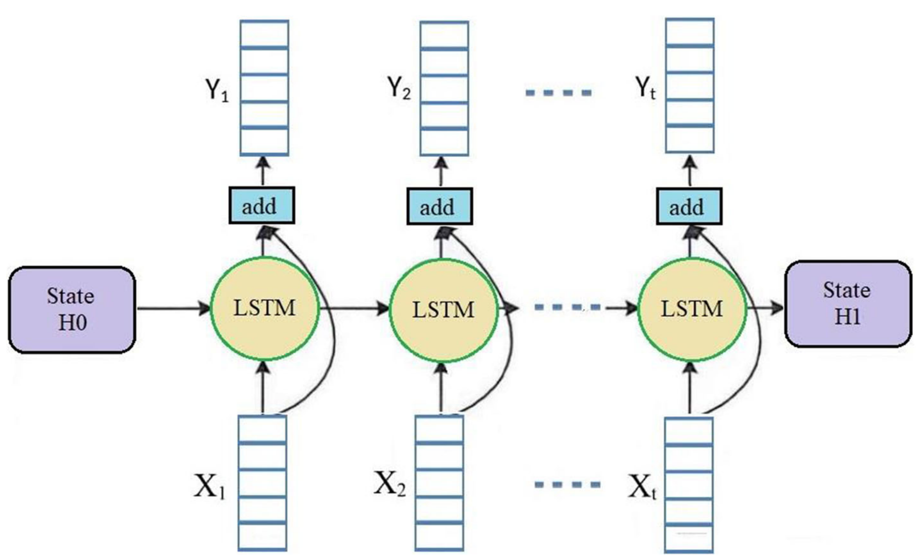

Fig. 6 RNN layer structure diagram

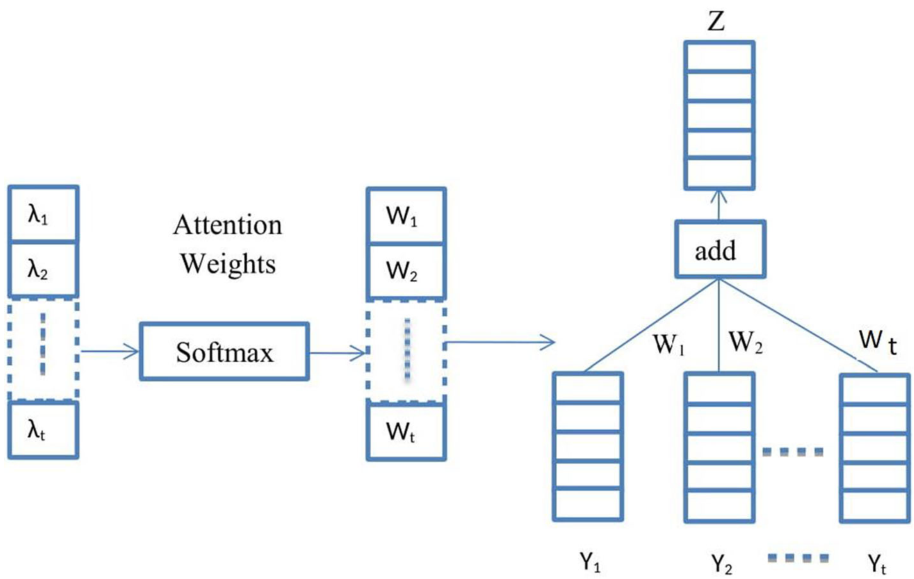

Fig. 7 Attention layer structure diagram

$$W = soft\ \max(\lambda) \quad (8)$$

$$h = W \otimes y \quad (9)$$

$$z = \sum_{i=1}^{t} h_i \quad (10)$$

where y is the input vector of the attention layer, z is the output vector of the attention layer, and $\otimes$ represents multiplication by element.

3.5 Performance measures

The last step after feature extraction and classification is to check the efficiency of the models in terms of various performance evaluation metrics that assists in tracking the performance of the model. In this study, the most common and important performance metrics namely accuracy, sensitivity, and specificity are computed with the help of confusion matrix. Another important performance measure, ROC curve, is used to measure the performance of the proposed system graphically. All the performance metrics are calculated by using the confusion table as shown in Table 1.

Following are the formulas through which these measures are computed:

Table 1 Confusion Matrix

	Predicted (−)	Predicted (+)
Actual (−)	TN	FP
Actual (+)	FN	TP

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \times 100 \quad (11)$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \times 100 \quad (12)$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100 \quad (13)$$

4 Experimental results and analysis

This section represents the experimental and simulation results attained via the two deep learning algorithms, i.e., LSTM and CRNN. The performance of the utilized deep learning models was checked on the athlete EEG dataset of the Stanford Research Project. The dataset is divided into two parts, i.e., training and testing, where 70% of data is used to train the models, while the remaining 30% is used for the validation purpose. To measure the performance of the utilized DL approaches, different performance metrics were used. Further, preprocessing techniques are also applied on all features before used by the classification algorithms.

4.1 Simulation results and analysis of LSTM architecture

This subsection illustrates the simulation results and experimental analysis of the LSTM model. Indeed hyper-parameters setting is a significant step in machine learning and deep learning algorithms for attaining the highest results. Various parameters were tuned to achieve best performance using the LSTM. The important parameters in setting the LSTM architecture are: batch-size, training epochs, size of hidden layers, recurrent depth, dropout size, and the L2 regularization coefficient of every LSTM layer. In addition, some other hyper parameters were also tuned for improving the performance of the model which includes the Adam optimizer in which the model performance was checked by tuning the learning rate and exponential decay rates. For the validation scheme, different set of parameters were assigned. For training the model, the loss function used in the proposed architecture was binary cross-entropy.

LSTM showed good performance by attaining the training accuracy of 93.40% and validation accuracy of 89.90%. The training and validation accuracies increase with the number of epochs, while the training and validation loss of the LSTM model decreases with the increase in epochs. Figure 8 illustrates the training and validation accuracies of the LSTM architecture, while Fig. 9 demonstrates the training and validation loss of the LSTM architecture.

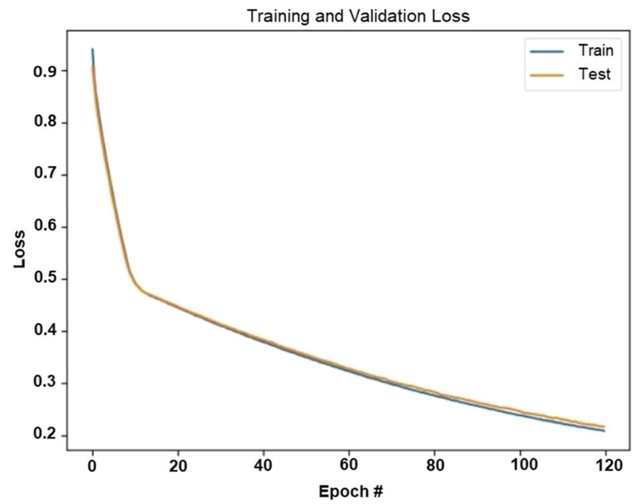

Fig. 9 Training and validation loss of the LSTM architecture

4.2 Simulation results and analysis of CRNN model

This subsection describes the simulation results and experimental analysis of the CRNN model that uses both the combination of Convolution and Recurrent Neural Network. For the CRNN model, multiple experiments were performed that can be categorized into the following subsections.

4.2.1 Experimental analysis of the feature layer using different sampling multiples

In order to determine the influence of different sampling multiples of the input EEG data at the feature layer for the recognition of EEG signal results, we did the following comparative experiments. 1) no sampling of the input data, 2) 2,4,and 8 times sampling of the input data. According to the experiments, best performance was observed at 2 times sampling. Specifically, for 2 time sampling we set the 4th residual channel attention unit of CAMResNet13 to perform 2 times sampling on the input (i.e., set the stride of the first convolutional layer of the residual unit to 2). For 4 times sampling, we set CAResNetl3 to sample the input by 2 times for every 3rd residual channel attention units starting from the first residual channel attention unit and perform 2 times sampling twice, i.e., a total of the input data are sampled 4 times. Further, for 8 times sampling, each 2 residual channel attention unit is set to sample the input twice, and the input data are sampled 8 times in total. The other parameters of CAMResNet13 are configured according to the best experimental performance. For example, the size of the convolution kernel ‘k’ is set to 5, and the compression ratio ‘r’ is set to 8. All the simulation results attained via performing multiple experiments are shown in Table 2.

It can be observed from the experimental results that when the other conditions remained unchanged and the input data are sampled 2 times and it shows good

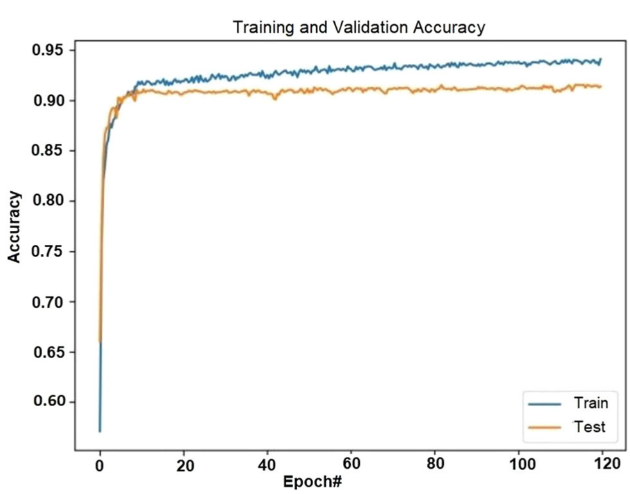

Fig. 8 Training and validation accuracy of the LSTM architecture

Table 2 Comparison of recognition results under different sampling multiples at the feature layer

Method	Rate of accuracy
CRNN + no sampling	89.98
CRNN + 2 times sampling	91.05
CRNN + 4 times sampling	87.93
CRNN + 8 times sampling	86.74

performance by recognizing the EEG signals in an efficient manner. Based on the experiments, the larger the sampling multiple is, the worse the effect will be, but the most ideal sampling multiple needs to be obtained through experiments based on a specific tasks.

In order to analyze the recognition effect of the network structure designed for each category, the confusion matrix is used as another indicator to measure the quality of the network model. Table 3 shows the confusion matrix of the proposed CRNN (2 times sampling) EEG signal recognition results for all the categories on the test set. From Table 3, it can be seen that CRNN's recognition results in each category have been greatly improved with CAMResNetl3 as compared to ResNet13, and there is no phenomenon that a certain recognition effect is particularly poor. In general, the network structure is for each category, and it achieved good recognition results.

4.2.2 Experimental analysis of attention layer

In order to verify whether the attention layer is effective for EEG signal recognition or not, based on the best results of the first set of experiments (CRNN + 2 times sampling), we removed the attention layer of the network (CRNN + 2 times sampling) and changed it to the direct network. The eigenvectors output at each time of the RNN layer was averagely weighted, and then the experiment was performed. The experimental result attained via such experiment is shown in Table 4. From Table 4, it is obvious that when the other conditions remained unchanged, the attention layer is very useful in the recognition of EEG signals.

In order to further analyze the contribution of each input vector of the attention layer to recognize the EEG signal, we visualize the weight coefficients of each input time vector of the attention layer in the CRNN (2 times sampling) network structure as shown in Fig. 10. The abscissa represents the weight coefficient corresponding to the input vector at each moment, and the ordinate represents the proportion of each weight coefficient, and the sum of these coefficients is always equal to 1. It can be seen from the histogram of the weighted coefficients that, the input vector at the later time of the attention layer contributes more to the EEG signal recognition task than the input vector at the previous time. As the input vector sequence of the attention layer comes from the output vector of the RNN layer each time, it can be considered that the output vector at the later time of the RNN layer contributes more than the output vector at the earlier time. This is also more in line with intuitive understanding. For RNN, the output feature vector at the later time encodes the information of the input vector at the previous time, so the degree of contribution should be greater.

Table 3 Confusion matrix of CRNN (2 times sampling) recognition

True label	Predicted label	
	HF	IO
HF	0.913	0.087
IO	0.092	0.908

4.2.3 Experimental analysis of classification layer

Finally, this article also compares the effect of different network structures (ResNet13, CAMResNet13) on the recognition results of EEG signals in the feature layer. It is also based on the best results of the first set of experiments (CRNN + 2 times sampling). The network (CRNN + 2 times sampling) feature layer structure CAMResNet13 is changed to ResNetl3 for experiment. The experimental results attained via this approach are shown in Table 5. From the experimental results, it can be seen that the network structure using CAMResNet13 as the feature layer is much better than ResNet13, which again proves that the channel attention module and network structure, i.e., CAMResNet13 proposed in this study is very effective and useful for the recognition of EEG signals.

Table 6 shows the overall performance of the proposed CRNN with 2 times sampling using two different architectures, i.e., CAMResNet13 and ResNet13 at the feature layer.

From Table 6, it is obvious that CRNN (2 times sampling) along with CAMResNet13 showed good performance in terms of all performance measures (accuracy, sensitivity, and specificity) by attaining the accuracy of 91.05%, 90.80% of sensitivity, and specificity of 91.30%.

Figure 11 illustrates the performance of the utilized two approaches, while Fig. 12 notifies the ROC curves of the proposed approaches.

We also analyze the channel weights in CRNN (2 times sampling CAMResNent3) to show the contribution of different channel of EEG signals to recognize the class of EEG signals. We observed that the channel weight vectors of different categories are somewhat different, but regardless of the category, their more important channel weights are basically between 60 and 80 channels. The simulation results also showed that the channel attention module used in this study can be used to establish a relationship between the channels.

Table 4 Comparison of recognition results with and without attention layer

Method	Rate of accuracy
CRNN + 2 times sampling (Baseline)	91.05
CRNN + 2 times sampling (Without attention layer)	88.56

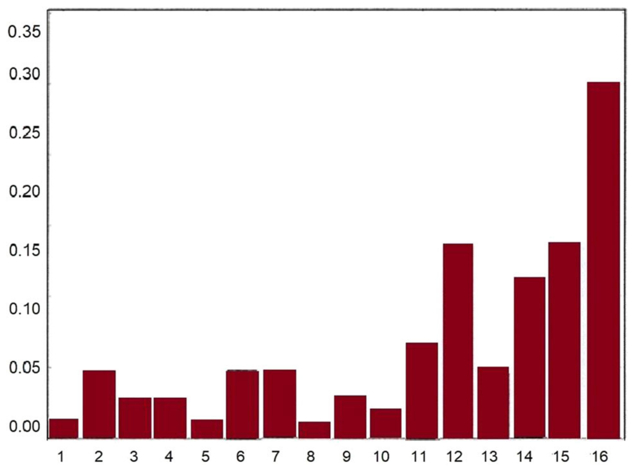

Fig. 10 Histogram of Attention Layer weighted coefficients

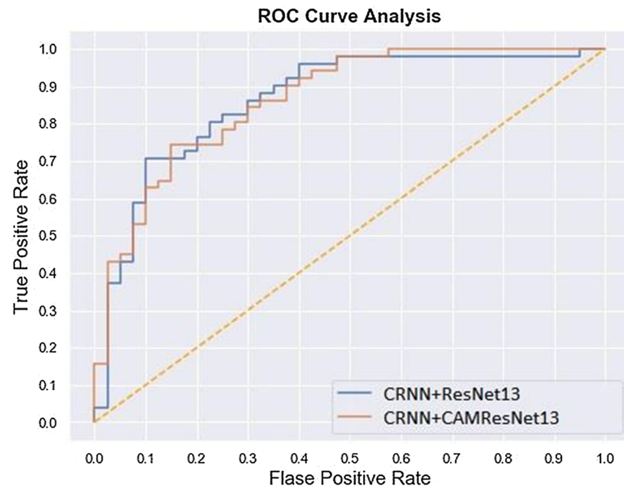

Fig. 12 ROC Curve analysis of the proposed techniques

Table 5 Comparison of recognition results of feature layers using different network structures

Method	Rate of accuracy
CRNN + 2 times sampling (CAMResNet13)	91.05
CRNN + 2 times sampling (ResNet13)	87.26

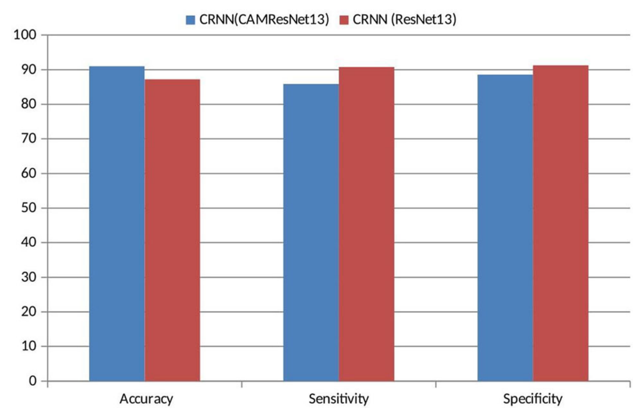

Fig. 11 Performance of the proposed CRNN + 2 times sampling using different architectures

5 Conclusions

To measure the health conditions of an athlete is an essential task to help him/her in continuing his/her career for a long time. This study mainly focuses on the EEG signal recognition methods based on two deep learning approaches, i.e., long short-term memory (LSTM) and convolutional recurrent neural network (CRNN). First, we analyze the EEG signals using the LSTM and got promising results, i.e., 89.90% validation accuracy. After that, the architecture and implementation process of the convolutional recurrent network was introduced in detail. Several experiments were performed using various sample sizes along with CRNN + ResNet13 and CRNN + CAMResNet13. Finally, the EEG dataset from the Stanford research project proved the effectiveness of our proposed method and explored the influence of some hyper parameters in the network on the recognition results. Our proposed method has much better results. CRNN + ResNet13 attained the recognition accuracy of 87.26%, and CRNN + CAMRestNet13 achieved an accuracy of 91.05%, sensitivity of 85.90%, and specificity of 88.59%, respectively. In the future, we aim to extend our approach for a much larger dataset with a massive number of features

Table 6 Performance of the proposed CRNN + 2 times sampling using different architectures

Method	Accuracy	Sensitivity	Specificity
CRNN + 2 times sampling (CAMResNet13)	91.05	90.80	91.30
CRNN + 2 times sampling (ResNet13)	87.26	85.90	88.59

to be extracted in sport informatics to provide numerous health benefits to athletes.

Acknowledgements This paper is supported by Humanities and Social Science Research Project of Henan Provincial

Department of Education Subject Number: 2021-ZZJH-200 and Special Fund for the Introduction and Training of High-level Talents of Henan University of Chinese Medicine: 00104311-2020-1.

Declarations

Conflict of interest All authors declare that all authors have no conflict of interest.

References

1. Ahangi A, Karamnejad M, Mohammadi N, Ebrahimpour R, Bagheri N (2013) Multiple classifier system for EEG signal classification with application to brain–computer interfaces. Neural Comput Appl 23(5):1319–1327
2. Gümüslü E, Erol Barkana D, Köse H (2020) October. Emotion recognition using EEG and physiological data for robot-assisted rehabilitation systems. In: Companion publication of the 2020 international conference on multimodal interaction, pp 379–387
3. Dash S, Acharya BR, Mittal M, Abraham A, Kelemen AG (eds) (2020) Deep learning techniques for biomedical and health informatics. Springer, Cham
4. Yang S, Li B, Zhang Y, Duan M, Liu S, Zhang Y, Feng X, Tan R, Huang L, Zhou F (2020) Selection of features for patient-independent detection of seizure events using scalp EEG signals. Comput Biol Med 119:103671
5. Jan MA, Khan F, Khan R, Mastorakis S, Menon VG, Watters P, Alazab M (2020) A Lightweight Mutual Authentication and Privacy-preservation Scheme for Intelligent Wearable Devices in Industrial-CPS. IEEE Trans Industr Inf. https://doi.org/10.1109/TII.2020.3043802
6. Jiancheng L, Zhanyu C (1999) prediction of EEG signal based on neural network. Appl Electron Technol 07:14–15
7. Mao X, Meng Q (2005) EEG classification based on wavelet transform and neural network. J Beijing Univ Aeron Astron
8. Cui Z (2019) Research on EEG signal analysis method based on convolution neural network
9. Guohang F, Bingmei S, Boyi W et al (2019) Application of artificial neural network in EEG signal processing. Electron world 568(10):54–55
10. Xungung M (2020) Recognition of multi class EEG signals based on convolutional neural network
11. Hongyu L, Qingjiang L, Xiaojuan C et al. (2019) Classification of EEG signals based on adaptive cascade neural network. Ind Innov Res
12. Li T, Zhong ZJ, Jian Z et al (2019) Emotion recognition of EEG signals based on improved convolutional neural network. Comput Eng Appl 055(022):99–105
13. Hongtao W, Hui H, Yuebang H et al (2019) Analysis method of event-related potential EEG signal based on noise reduction self-coding neural network. Control Theory Application 036(004):589–595
14. Jingxia C, Liyan W, Xiaoyun J et al (2019) Emotion recognition of EEG signals based on deep convolution neural network. Comput Eng Appl 55(937):109–116
15. Yuetian L (2019) Application of convolutional neural network based on large interval softmax loss function in EEG signal classification
16. Xianlun T, Wei Li, Weichang Ma et al (2020) EEG recognition based on conditional empirical mode decomposition and serial parallel CNN. Acta Electron Sin 042(004):1041–1048
17. Subasi A (2007) EEG signal classification using wavelet feature extraction and a mixture of expert model. Expert Syst Appl 32(4):1084–1093
18. Jiang X, Wei S, Zhang S, Zhao L, Liu F, Tang H, He Z, Liu C (2019) Comparison of photoplethysmographic signal features between healthy and sleep apnea patients during five sleep stages. J Med Imag Health Inf 9(1):63–69
19. Zhang J, Yang L, Su Z, Mao X, Luo K, Liu C (2018) Photoplethysmogram signal quality assessment using support vector machine and multi-feature fusion. J Med Imag Health Inf 8(9):1757–1762
20. Liu F, Liu C, Zhao L, Zhang X, Wu X, Xu X, Liu Y, Ma C, Wei S, He Z (2018) An open access database for evaluating the algorithms of electrocardiogram rhythm and morphology abnormality detection. J Med Imag Health Inf 8(7):1368–1373
21. Liu Y, Wei S, Zhang S, Liu F, Li J, Liu C (2018) Signal quality index based two-step method for heart rate estimation by combining electrocardiogram and arterial blood pressure signals. Journal of Medical Imaging and Health Informatics 8(7):1502–1507
22. Lu S, Lu Z, Phillips P, Wang S, Zhang Y (2018) Pathological brain detection in magnetic resonance imaging using combined features and improved extreme learning machines. J Med Imag Health Inf 8(7):1486–1490
23. Zhao Z, Liu C, Li Y, Li Y, Wang J, Lin BS, and Li J (2019) Noise rejection for wearable ECGs using modified frequency slice wavelet transform and convolution neural networks. In: IEEE Access
24. Xu Yu, Zhan D, Liu L, Lv H, Lingwei Xu, Junwei Du (2021) A privacy-preserving cross-domain healthcare wearables recommendation algorithm based on domain-dependent and domain-independent feature fusion. IEEE J Biomed Health Inform. https://doi.org/10.1109/JBHI.2021.3069629
25. Xu Yu, Chu Y, Jiang F, Guo Y, Gong D (2018) SVMs Classification based two-side cross domain collaborative filtering by inferring intrinsic user and item features. Knowl Based Syst 141:80–91
26. Li Y, Wen PP (2014) Modified CC-LR algorithm with three diverse feature sets for motor imagery tasks classification in EEG based brain-computer interface. Comput Methods Prog Biomed 113(3):767–780
27. Ashok A, Bharathan AK, Soujya V, Nandakumar P (2013) Tikhonov regularized spectrally weighted common spatial patterns. In 2013 International conference on control communication and computing (ICCC), IEEE, pp 315–318
28. Ang KK, Chin ZY, Wang C, Guan C, Zhang H (2012) Filter bank common spatial pattern algorithm on BCI competition IV datasets 2a and 2b. Front Neurosci 6:39
29. Tabar YR, Halici U (Feb 2017) A novel deep learning approach for classification of EEG motor imagery signals. J Neural Eng 14(1):016003
30. Lee HK, Choi Y-S, (2018) A convolution neural networks scheme for classification of motor imagery EEG based on wavelet time-frequecy image. In 2018 International conference on information networking (ICOIN), IEEE, pp 906–909
31. Naseer N, Hong K-S (2013) Classification of functional near-infrared spectroscopy signals corresponding to the right-and left-wrist motor imagery for development of a brain-computer interface. Neurosci Lett 553:84–89
32. Siuly S, Li Y (2012) Improving the separability of motor imagery EEG signals using a cross correlation-based least square support vector machine for brain-computer interface. IEEE Trans Neural Syst Rehabil Eng 20(4):526–538

33. Yang H, Sakhavi S, Ang KK, Guan C (2015) On the use of convolutional neural networks and augmented CSP features for multi-class motor imagery of EEG signals classification. In 2015 37th Annual international conference of the ieee engineering in medicine and biology society (EMBC), IEEE, pp 2620–2623
34. Miao Yu, Quan T, Qinglong Peng XuYu, Liu L (2021) A model-based collaborate filtering algorithm based on stacked autoencoder. Neural Comput Appl. https://doi.org/10.1007/s00521-021-05933-8